IN THE CLAIMS

Please amend claims 5, 15, and 31 as follows below.
The following is a complete listing of claims.

MARKED-UP LISTING OF CLAIMS

1. (Original) A design method for transforming sequential logic designs into equivalent combinational logic, the design method comprising:

simulating each stage of a clocking sequence to produce simulation values;

saving the simulation values; and

performing a plurality of backward logic traces based on the saved simulation values to provide an equivalent combinational logic representation of a sequential logic design.

2. (Original) The design method of claim 1 wherein for each time "T" of the clocking sequence, each block being measured at the time T is traced.

3. (Original) The design method of claim 2 wherein for each block being measured one or more items selected from a list comprising measure primary output and scan unload event are traced.

4. (Original) The design method of claim 1 further including simulating scan operations by placing the design in its scan state.

5. (Currently Ameneded) A ~~[[The]]~~ design method for transforming sequential logic designs into equivalent combinational logic, the design method comprising: ~~of claim 1 further including~~

simulating each stage of a clocking sequence to produce simulation values;

saving the simulation values;

simulating a scan operation; and

performing a plurality of backward logic traces based on the saved simulation values to provide an equivalent combinational logic representation of a sequential logic design.

6. (Original) The design method of claim 5 further including setting scan control inputs to their scan-enable values.

7. (Original) The design method of claim 5 further including turning off all clocks for the simulating stage.

8. (Original) The design method of claim 1 further including turning off all other clocks when a given clock is pulsed.

9. (Original) The design method of claim 1 further including simulating primary input force events by turning off all of the clock inputs to the design.

10. (Original) The design method of claim 1 wherein the design method is used for automatic test pattern generation.

11. (Original) The design method of claim 1 further including partitioning the design to be tested into a plurality of smaller pieces.

12. (Original) The design method of claim 11 wherein the plurality of smaller pieces are processed on separate computing devices.

13. (Original) The design method of claim 1 wherein the sequential design includes one or more chopped clocks.

14. (Original) The design method of claim 13 wherein, during the simulation of the clocking sequence, each distinct state of any chopped clocks is saved as a separate time frame.

15. (Currently Amended) The ~~A back tracing~~ design method of claim 1 ~~for transforming sequential logic designs into equivalent combinational logic~~, ~~the method comprising~~ wherein performing a plurality of backward logic traces further comprises:

determining whether a time T is negative;

if it is determined that the time T is negative, producing a block B at time T as a block tied to an unknown logic;

if it is determined that the time T is not negative, determining whether the block B has a known simulation value at time T; and

if it is determined that the block B has a known simulation value at time T, producing a tied block B at time T with the known simulation value.

16. (Original) The method of claim 15 wherein the known simulation value is selected from a group comprising 0, 1, and Z.

17. (Original) The method of claim 15 further including:

if it is determined that the block B has no known simulation value at time T, determining whether the block B is a combinational block;

if the block B is a combinational block, producing block B at time T with the same function as block B; and

for each input I of the block B, back tracing the design.

18. (Original) The method of claim 17 wherein the combinational block is selected from a group comprising an AND gate and an OR gate.

19. (Original) The method of claim 15 further including:

if it is determined that the block B has no known simulation value at time T, determining whether the block B is a latch; and

if it is determined that the block B is a latch, processing the block B as a latch.

20. (Original) The method of claim 15 further including:

if it is determined that the block B has no known simulation value at time T, determining whether the block B is a primary input; and

if it is determined that the block B is a primary input, processing the block B as a primary input.

21. (Original) An article of manufacture for transforming sequential logic designs into equivalent combinational logic, the article comprising:

a machine readable medium that provides instructions that, if executed by a machine, will cause the machine to perform operations including:

simulating each stage of a clocking sequence to produce simulation values;

saving the simulation values; and

performing a plurality of backward logic traces based on the saved simulation values to provide an equivalent combinational logic representation of a sequential logic design.

22. (Original) The article of claim 21 wherein for each time "T" of the clocking sequence, each block being measured at the time T is traced.

23. (Original) The article of claim 22 wherein for each block being measured one or more items selected from a list comprising measure primary output and scan unload event are traced.

24. (Original) The article of claim 21 wherein the operations further include simulating scan operations by placing the design in its scan state.

25. (Original) The article of claim 21 wherein the operations further include simulating primary input

force events by turning off all of the clock inputs to the design.

26. (Original) The article of claim 21 wherein the operations are used for automatic test pattern generation.

27. (Original) The article of claim 21 wherein the operations further include partitioning the design to be tested into a plurality of smaller pieces.

28. (Original) The article of claim 27 wherein the plurality of smaller pieces are processed on separate computing devices.

29. (Original) The article of claim 21 wherein the sequential design includes one or more chopped clocks.

30. (Original) The article of claim 21 wherein, during the simulation of the clocking sequence, each distinct state of the chopped clocks is saved as a separate time frame.

31. (Currently Amended) The [[An]] article of claim 21, wherein the operation of performing a plurality of backward traces further comprises the operations of ~~manufacture transforming sequential logic designs into equivalent combinational logic, the article comprising:~~

~~a machine readable medium that provides instructions that, if executed by a machine, will cause the machine to perform operations including~~:

determining whether a time T is negative;

if it is determined that the time T is negative, producing a block B at time T as a block tied to an unknown logic;

if it is determined that the time T is not negative, determining whether the block B has a known simulation value at time T; and

if it is determined that the block B has a known simulation value at time T, producing a tied block B at time T with the known simulation value.

32. (Original) The article of claim 31 wherein the operations further include:

if it is determined that the block B has no known simulation value at time T, determining whether the block B is a combinational block;

if the block B is a combinational block, producing block B at time T with the same function as block B; and

for each input I of the block B, back tracing the design.

33. (Original) The article of claim 31 wherein the operations further include:

if it is determined that the block B has no known simulation value at time T, determining whether the block B is a latch; and

if it is determined that the block B is a latch, processing the block B as a latch.

34. (Original) The article of claim 31 wherein the operations further include:

if it is determined that the block B has no known simulation value at time T, determining whether the block B is a primary input; and

if it is determined that the block B is a primary input, processing the block B as a primary input.